DB2 for z/OS and DASD-Based Disaster Recovery

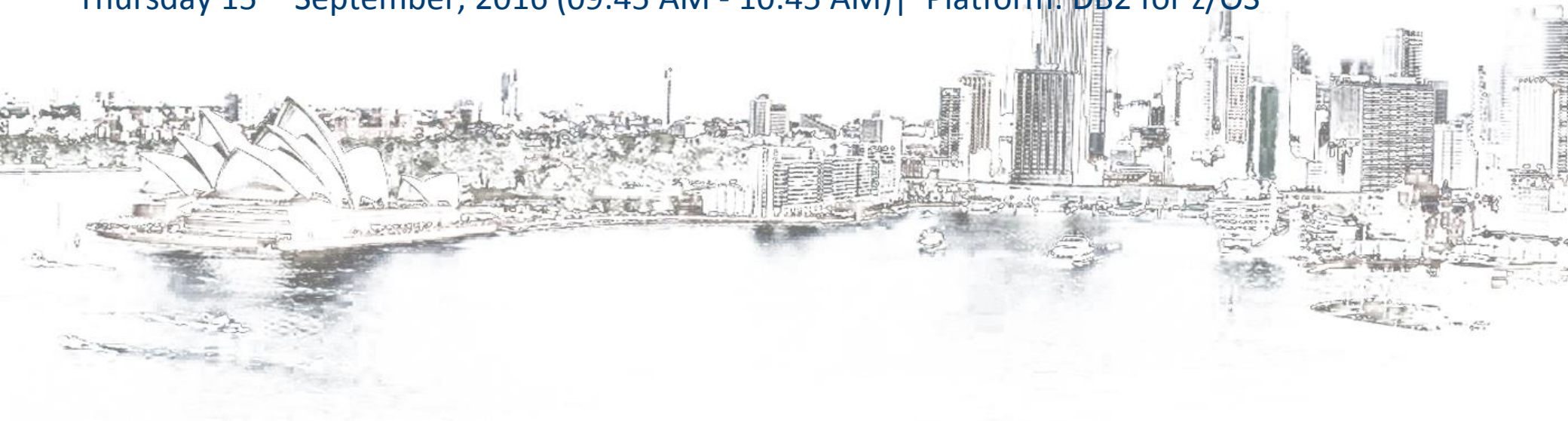
Blowing Away The Myths

Florence Dubois

IBM DB2 for z/OS Development

Session Code: M7

Thursday 15th September, 2016 (09:45 AM - 10:45 AM)| Platform: DB2 for z/OS



Agenda

- **Introduction**
- **DB2 Disaster Recovery before and after DASD-mirroring**
- **IBM Remote Mirror and Copy functions**
  - Metro Mirror
  - z/OS Global Mirror
  - Global Copy
  - Global Mirror
  - 3-site solutions
- **DB2 restart recovery**
  - Tune for fast restart
  - Optimise GRECP/LPL recovery
- **Testing**
- **Conclusion**

Introduction

- **Everything should start with the business objectives**

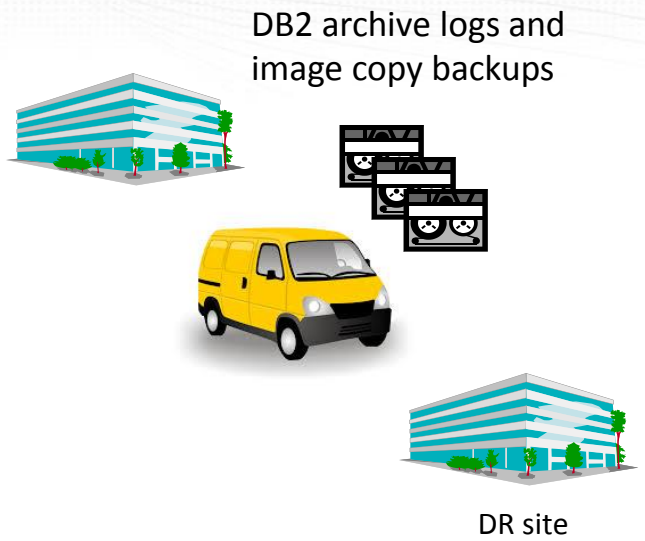
Introduction

- **Everything should start with the business objectives …**
  - 'Quality of Service' requirements for applications
    - Availability
      - High availability? Continuous operations? Continuous availability?
      - Restart quickly? Mask failures?
    - Performance
  - In case of a Disaster
    - *Recovery Time Objective (RTO)*
      - How long can the business afford to wait for IT services to be resumed after a disaster?
      - DB2 is only one of the component
    - *Recovery Point Objective (RPO)*
      - What is the acceptable time difference between the data in your production system and the data at the recovery site (consistent copy)?
      - In other words, how much data is your company willing to lose following a disaster?
  - Need to understand the real business requirements and expectations
    - These should drive the infrastructure, not the other way round

DB2 Disaster Recovery before and after DASD-mirroring

DB2 DR procedures with the traditional 'PTAM' method

Create ICF user catalogs
Restore DB2 libraries
Define active logs, BSDS, DB2 catalog and DB2 directory
Change the ZPARM needed for disaster recovery
Recover the BSDS
Start DB2 (conditional restart)
Resolve indoubt units of recovery
Recover the DB2 catalog and directory
Define and initialize the work file database
Modify ZPARM to restart all databases
Stop and start DB2
Make a full image copy of the DB2 catalog and directory
Recover user data and rebuild indexes
Make a full image copy of all TS and IX with COPY YES



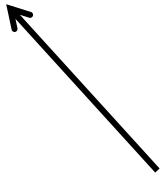
RPO depends on how often the archive logs are shipped to the DR site (if once a day, RTO=24 hours)

RTO = many hours to several days

DB2 Disaster Recovery before and after DASD-mirroring …

DB2 DR procedures with DASD-mirroring (non-data sharing)

Start DB2 (normal 'warm' restart)



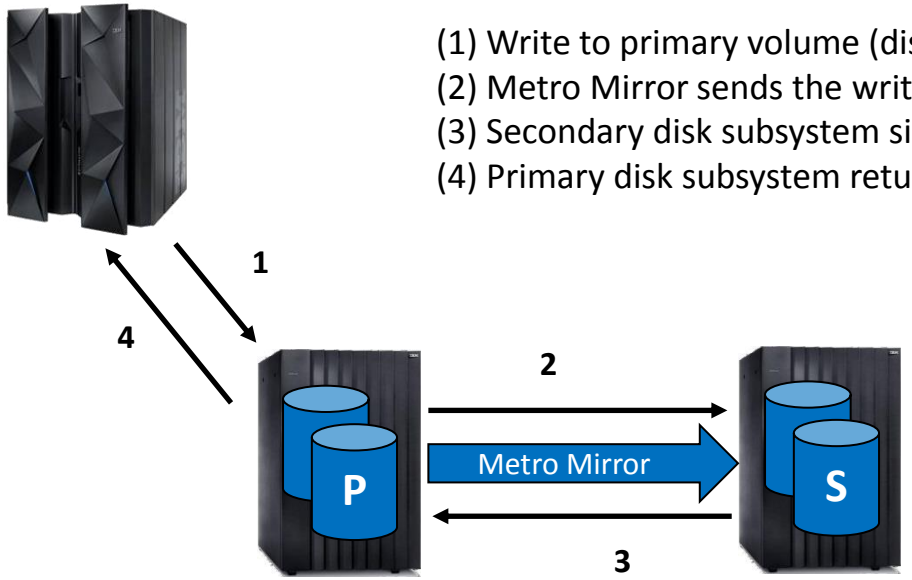
RPO = 0 to a few minutes
RTO = minutes to small number of hours

DB2 Disaster Recovery before and after DASD-mirroring …

- ***Dependent writes***
  - Critical concept for disk-based replication solutions
  - The start of a write operation is dependent upon the completion of a previous write to a disk in the same storage subsystem or a different storage subsystem
    - For example, typical sequence of write operations for a database update transaction:
      1. An application makes an update and the data page is updated in the buffer pool
      2. The application commits and the log record is written to the log
      3. At a later time, the update to the table space is externalized to disk
      4. A diagnostics log record is written to mark that the page has been externalized successfully
- **Data consistency for secondary DASD copy = *I/O consistency (crash consistency)***
  - Order of the dependent writes is preserved
  - Provides the capability to perform a database restart rather than a database recovery
    - Restart can be measured in minutes while recovery could be hours or even days
- **DB2 data consistency = *application consistency***
  - Re-established through normal DB2 warm restart recovery mechanisms
  - Requires an I/O consistent base

Metro Mirror

- **a.k.a. PPRC (Peer-to-Peer Remote Copy)**
- **Disk-subsystem-based synchronous replication**
  - z Systems and distributed data
  - Supported by IBM disk subsystems and other vendors



(1) Write to primary volume (disk subsystem cache and NVS)
(2) Metro Mirror sends the write I/O to secondary volume (cache and NVS)
(3) Secondary disk subsystem signals write complete
(4) Primary disk subsystem returns I/O complete

- **Limited distance**
  - Over long distances, performance impact on production running at the primary site

Metro Mirror …

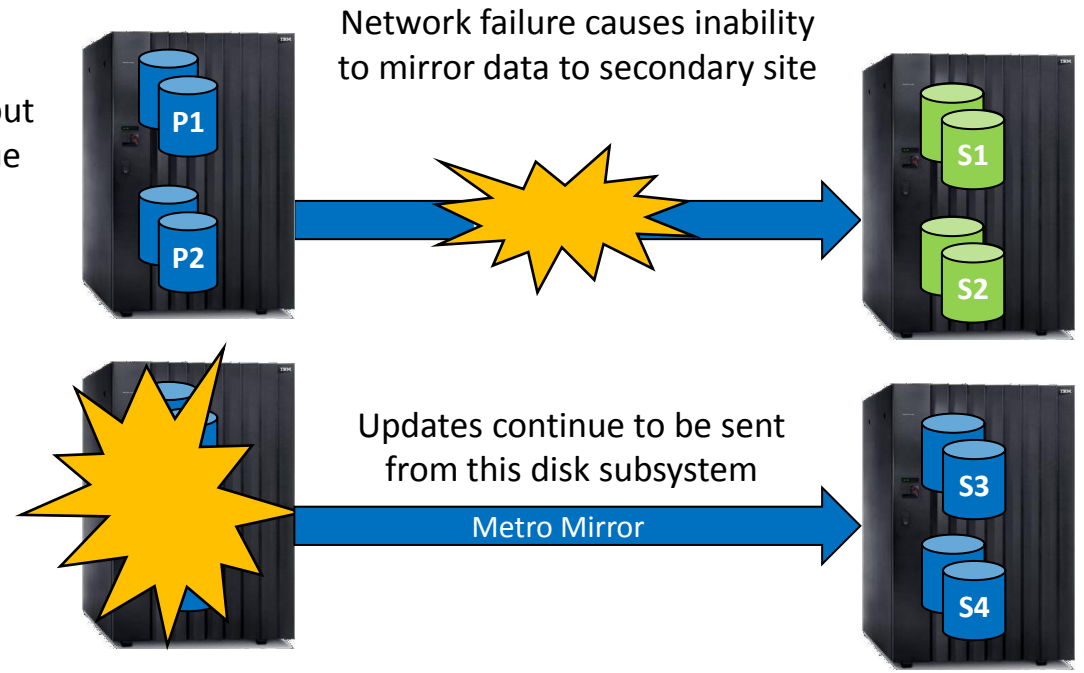
- **Myth #1: Sync replication always guarantees data consistency of the remote copy**
- **Answer: Not by itself…**
  - Metro Mirror operates at the device level (like any other DASD replication function)
    - Volume pairs are always consistent
  - But cross-devices and cross-boxes consistency is not guaranteed
    - An external management method is required to maintain consistency

Metro Mirror …

- **Traditional example of multiple disk subsystems**
  - In a real disaster (fire, explosion, earthquake), you can not expect your complex to fail at the same moment. Failures will be intermittent, gradual, and the disaster will occur over seconds or even minutes. This is known as the *Rolling Disaster*.

Volume pairs are suspended but writes are allowed to continue on primary devices (*)

(*) *Volume pairs are defined with CRIT(NO) - Recommended*

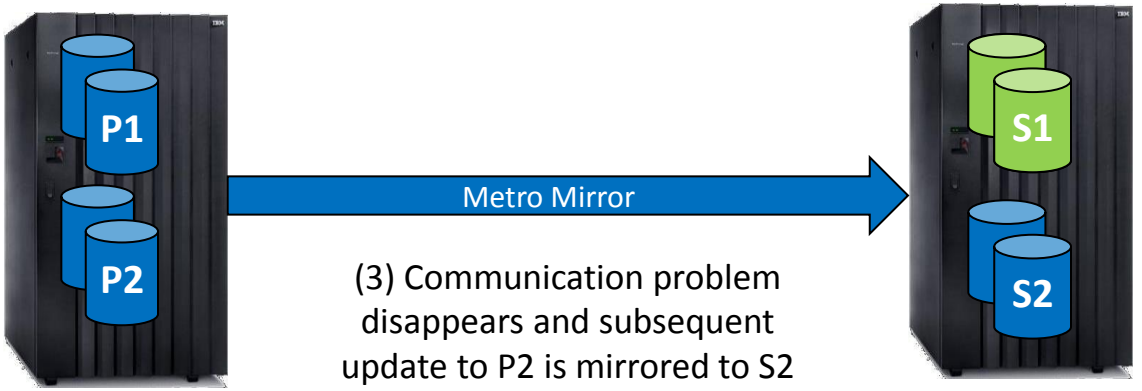


Devices on secondary disk subsystems are not consistent

Metro Mirror …

- **Example of single disk subsystem with intermittent problem with communication links or problem with the secondary DASD configuration**

(1) Temporary communications problem (e.g. network or SAN event) causes P1-S1 pair to suspend



(3) Communication problem disappears and subsequent update to P2 is mirrored to S2

(2) During this time no I/O occurs to P2 so P2-S2 pair remains in full duplex

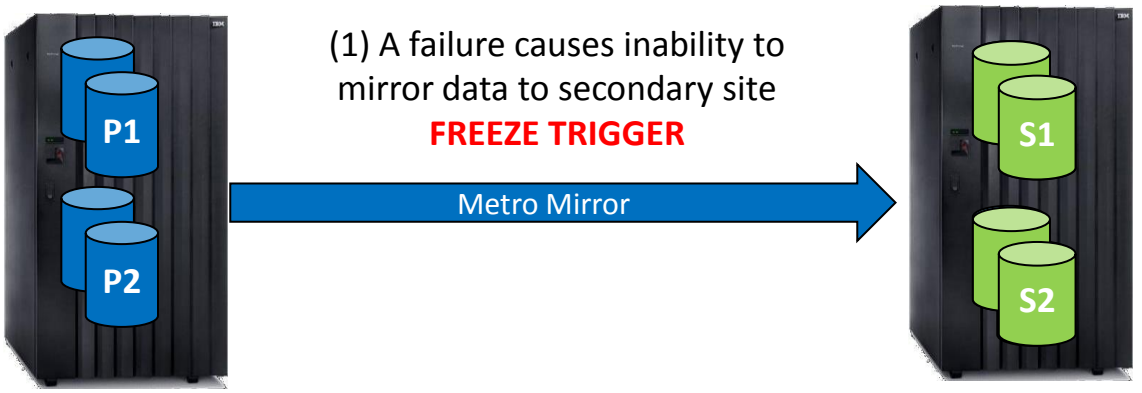
(4) S1 and S2 are now not consistent and if the problem was the first indication of a primary disaster we are not recoverable

Metro Mirror …

- **Consistency Group function combined with external automation**

(2) Volume pair that first detects the error is defined with CGROUP(Y): P1-S1 is suspended and P1 goes into an 'extended long busy' state - IEA494I is issued

(3) Automation is used to detect the alert and issue the CGROUP FREEZE command(*) to all LSS pairs that have volumes related to the application



(*) or equivalent

(5) Automation issues the CGROUP RUN(*) command to all LSS pairs to release the long busy - secondary devices are still suspended at a consistent point-in-time

(4) CGROUP FREEZE(*) suspends the volume pairs for each LSS, puts all primary devices in long busy state, and deletes the Metro Mirror paths

Metro Mirror …

- **Myth #1: Sync replication always guarantees data consistency of the remote copy**
- **Answer: Not by itself …**
  - Need to use the Consistency Group function AND external automation
    - Ensure that the whole environment is suspended as soon as a mirroring problem is detected, so that all the secondary devices are always consistent with each other
    - Solution must support both planned and unplanned situations
  - IBM offers services and solutions for the automation and management of a Metro Mirror environment including:
    - GDPS (Geographically Dispersed Parallel Sysplex)
      - GDPS/PPRC or GDPS/PPRC HM (HyperSwap Manager)
    - Tivoli Storage Productivity Center for Replication (full edition)
      - TPC-R Basic Edition with Basic HyperSwap does not include the capability to maintain recovery data consistency (no FREEZE capability) → not a disaster recovery solution

Metro Mirror …

- **Myth #2: Sync replication guarantees zero data loss in a disaster (RPO=0)**
- **Answer: Not by itself…**
  - The only way to ensure zero data loss is to immediately stop all update activity to the primary disks as soon as a mirroring problem is detected
    - e.g. if you lose connectivity between the primary and secondary devices
    - Requires automation to reset the production systems while I/O is suspended
      - GDPS PPRCFAILURE=STOP policy (Freeze and Stop)
      - GDPS PPRCFAILURE=COND policy (Freeze and Stop, conditionally)
        - If GDPS can determine that the freeze was triggered as a result of a secondary disk subsystem problem (no risk of a local disaster), GDPS performs a GO – otherwise, GDPS performs a STOP
  - Choosing to have zero data loss really means that
    - You have automation in place that will stop all I/O activity in the appropriate circumstances
    - You accept a possible impact on continuous availability at the primary site
      - Systems could be stopped for a reason other than a real disaster (e.g. broken remote copy link rather than a fire in the computer room)
      - GDPS PPRCFAILURE=COND policy can be used to minimise the ‘false positives’

Metro Mirror …

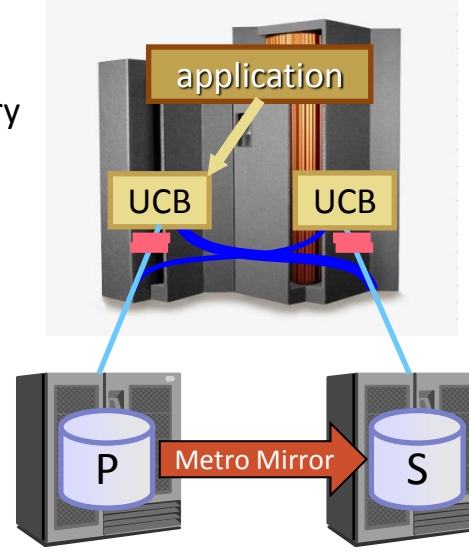
- **Myth #3: Metro Mirror eliminates DASD subsystem as single point of failure**
- **Answer: Not by itself…**
  - Need connectivity between host systems and the secondary disk subsystems
  - AND a non-disruptive failover HyperSwap capability e.g.,
    - GDPS/PPRC Hyperswap Manager (GDPS/HM) or GDPS/PPRC
    - z/OS Basic Hyperswap in TPC-R Basic Edition
    - Tivoli Storage Productivity Center for Replication for System z (full edition)

1) A permanent I/O error on a primary volume is detected

HYPERSWAP TRIGGER

2) Validate that the SWAP can be done

3) I/O quiesced and mirroring suspended to ensure data consistency

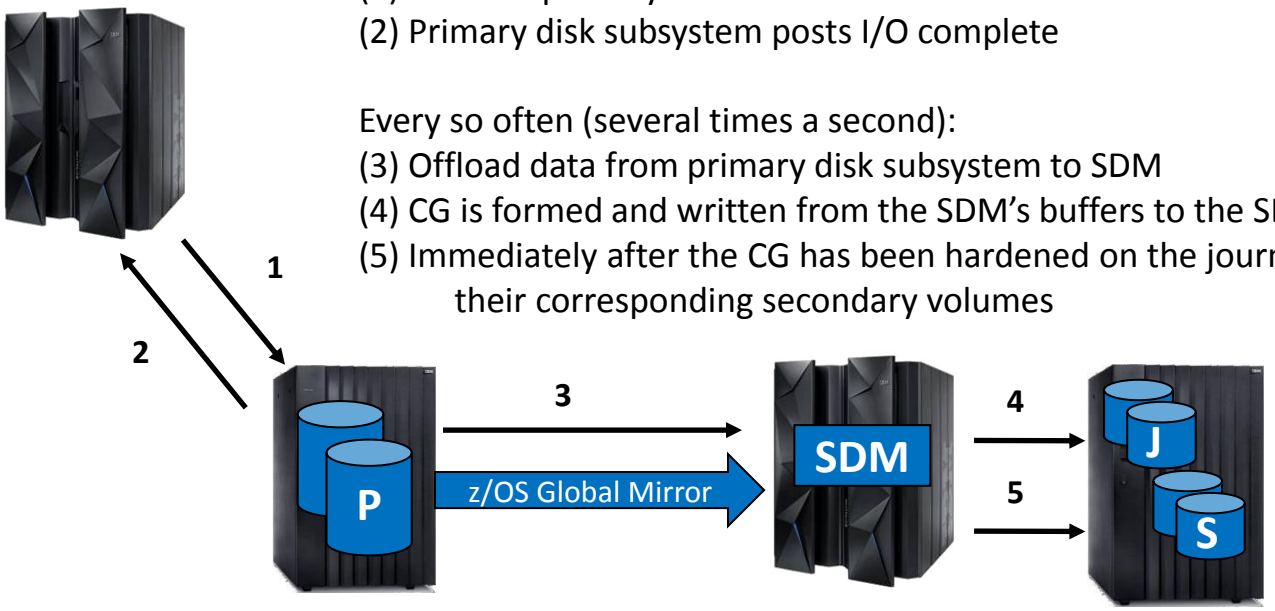


4) UCBs swapped on all systems in the sysplex (logical switch) and I/O resumed

3) Secondary devices are made available (PPRC configuration is physically switched)

z/OS Global Mirror

- **a.k.a. XRC (eXtended Remote Copy)**
- **Combination of hardware and software functions for asynchronous replication**
  - Involves disk subsystem microcode + a System Data Mover (SDM) in z/OS DFSMSdfp
  - Solution unique to z Systems data (z/OS, z/VM, Linux on z Systems)
  - Supported by IBM disk subsystems and other vendors



(1) Write to primary volume
(2) Primary disk subsystem posts I/O complete

Every so often (several times a second):
(3) Offload data from primary disk subsystem to SDM
(4) CG is formed and written from the SDM's buffers to the SDM's journal data sets
(5) Immediately after the CG has been hardened on the journal data sets, the records are written to their corresponding secondary volumes

z/OS Global Mirror …

- **Use of Time-stamped Writes and Consistency Groups to ensure data consistency**
  - All records being written to z/OS Global Mirror primary volumes are time stamped
  - Consistency Groups are created by the SDM
    - A Consistency Group contains records that the SDM has determined can be safely written to the secondary site without risk of out-of-sequence updates
    - Order of update is preserved across multiple disk subsystems in the same zGM session

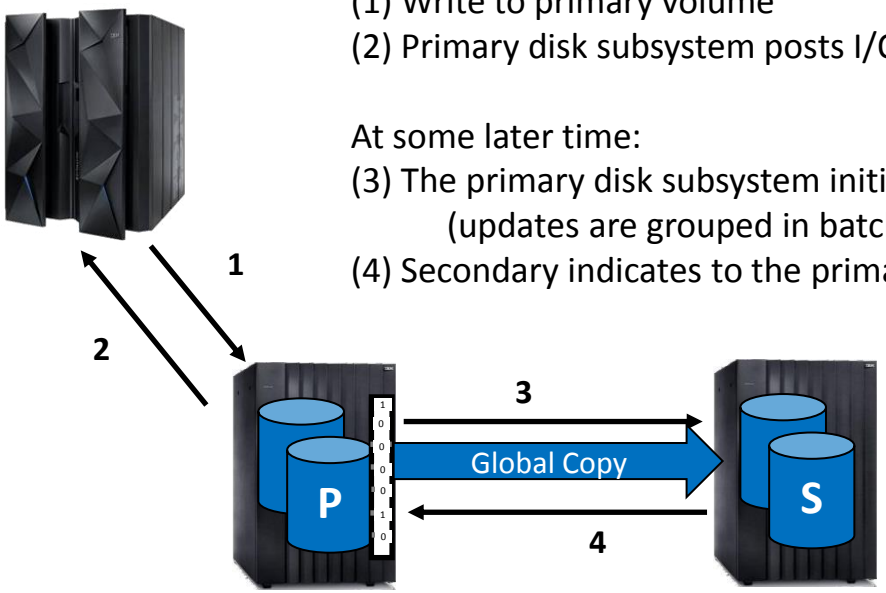
- **Recovery Point Objective**
  - Amount of time that secondary volumes lag behind the primary
  - Design point: RPO less than two minutes, typically 3-5 seconds
  - Depends mainly on
    - Performance of the SDM (MIPS, storage, I/O configuration)
    - Amount of bandwidth
    - Use of device blocking or write pacing

z/OS Global Mirror …

- **Myth #4: Because it is async, zGM has no impact on the primary devices**
- **Answer: It might… if you have enabled device blocking or write pacing**
  - Objective of these features: avoid SDM overload and help maintain a guaranteed RPO
    - Device blocking: Pause I/O write activity for volumes that have accumulated a large number of updates in the cache
    - Write pacing: Slow down I/O write activity by injecting small variable delays
  - Concerns:
    - Excessive pacing can result in unacceptable application throughput and response time
    - Insufficient pacing can result in extended long busy conditions
    - In both cases, increased risk of system slowdowns
  - Recommendations:
    - Do not use device blocking
    - Should not use aggressive write pacing on DB2 volumes, especially not the DB2 active logs
    - Suspend the zGM session instead of setting extended long busy (SuspendOnLongBusy=YES)
    - Carefully monitor pacing: if persistent, excessive write pacing is observed, analyse XRC performance data to determine the underlying cause and take remedial actions

Global Copy

- **a.k.a. PPRC-XD (Peer-to-Peer Remote Copy Extended Distance)**
- **Disk-subsystem-based asynchronous replication**
  - Primarily designed for remote data migration, transmission of inactive database logs, or off-site backups



(1) Write to primary volume
(2) Primary disk subsystem posts I/O complete

At some later time:
(3) The primary disk subsystem initiates transmits the data to the secondary disk subsystem
(updates are grouped in batches for efficient transmission)
(4) Secondary indicates to the primary that the write is complete - primary resets OOS bitmap

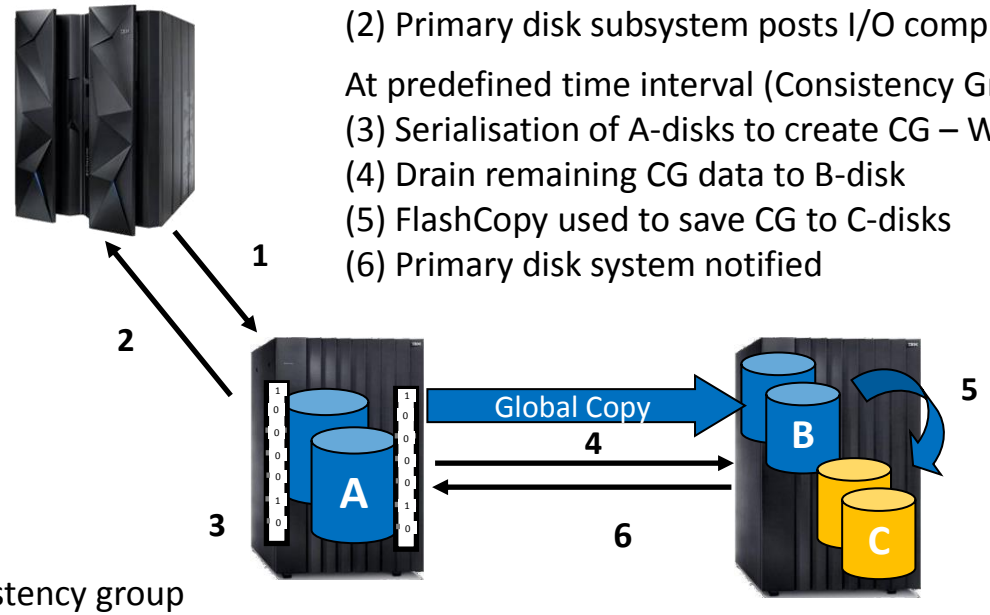
Out-of-sync (OOS) bitmaps

Global Copy …

- **Myth #5: Global Copy provides a remote copy that would be usable in a disaster**
- **Answer: Not by itself…**
  - Global Copy does NOT guarantee that the arriving writes at the local site are applied to the remote site in the same sequence
    - Secondary copy is a 'fuzzy' copy that is just not consistent
  - To create a consistent point-in-time copy:
    - Quiesce all updates to the primaries (e.g. use -SET LOG SUSPEND for DB2 data)
    - Perform the catch up by doing a go-to-sync operation (or wait until all updates have been transmitted y the secondary site)
    - 'Freeze' (suspend) the Global Copy pairs after they reach the Full Duplex state
    - Update activity on the primaries can be resumed (e.g. use -SET LOG RESUME for DB2 data)
    - FlashCopy the secondary volumes to create a backup copy with data consistency
    - Re-establish suspended pairs (resync)

Global Mirror

- **Combines Global Copy and FlashCopy**
- **Disk-subsystem-based asynchronous replication WITH consistency**
  - z Systems and distributed data
  - Supported by IBM disk subsystems



(1) Write to primary volume
(2) Primary disk subsystem posts I/O complete

At predefined time interval (Consistency Group Interval time (CGI) – default=zero seconds):
(3) Serialisation of A-disks to create CG – Write I/Os queued for short period of time (~2-3 ms)
(4) Drain remaining CG data to B-disk
(5) FlashCopy used to save CG to C-disks
(6) Primary disk system notified

Consistency group co-ordination and formation (change recording bitmaps)

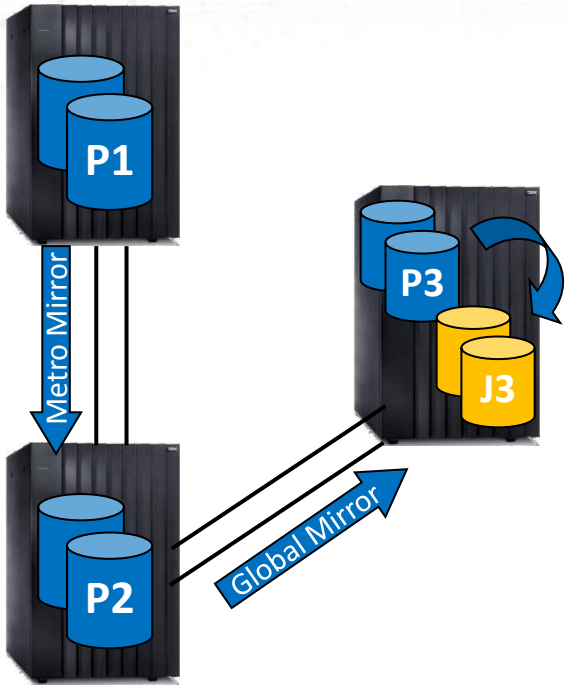
Data transmission

Consistency group save

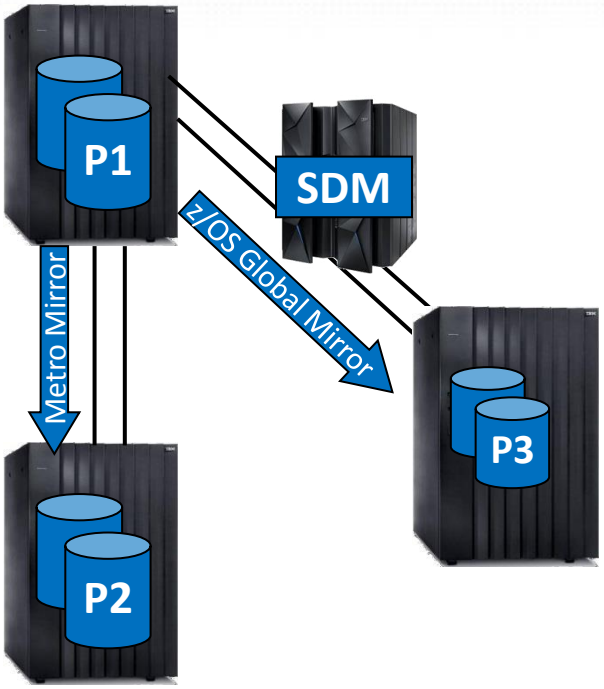
Global Mirror …

- **Recovery Point Objective**
  - Amount of time that FlashCopy target volumes lag behind the primary
  - Design point: RPO as low as 5 seconds
  - Depends mainly on
    - Load on the primary disk subsystems
    - Bandwidth and links between primary and secondary disk subsystems
    - Distance/latency between primary and secondary
    - Hotspots on secondary in write intensive environments
  - No pacing mechanism
    - Designed to protect production performance at the expense of the mirror currency
    - RPO can increase significantly if production write rates exceed the available resources

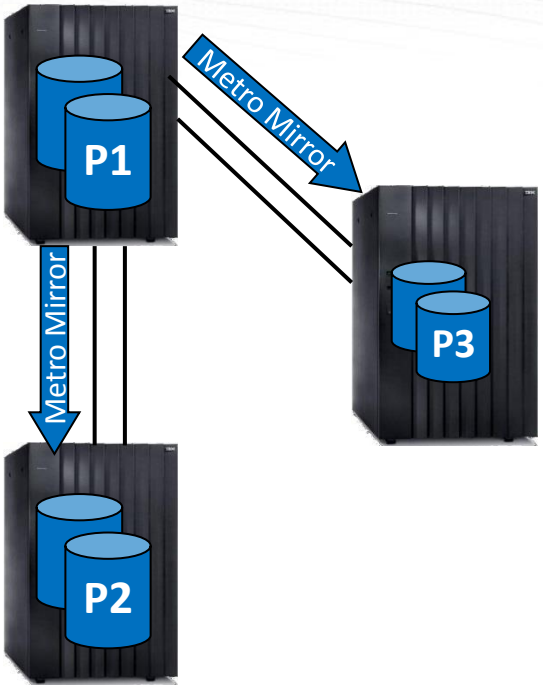
3-Site Solutions



Metro/Global Mirror (MGM)



z/OS Metro/Global Mirror (zMGM)



Multi-Target Metro Mirror
**** New with DS8870 Release 7.4 ****

DB2 Restart Recovery

- **DB2 restart is the key ingredient for disk-based DR solutions**
  - Normal DB2 warm restart
    - Re-establishes DB2 data consistency through restart recovery mechanisms
    - Do not use RESTART LIGHT
      - DB2 Restart Light will be slower than normal restart
      - Data Sharing: will not remove all retained locks and no automatic GRECP recovery
      - DB2 Restart Light should only be used for cross-system restart after LPAR failure
  - DB2 restart times have a direct impact on the ability to meet the RTO
- **DB2 restart MUST NOT be intended on a mirrored copy that is not consistent**
  - Guaranteed inconsistent data that will have to be fixed up
  - No way to estimate the damage
  - After the restart it is too late if the damage is extensive
  - Damage may be detected weeks and months after the event
- **DB2 cold start or any form of conditional restart WILL lead to data corruption and loss of data**

DB2 Restart Recovery …

- **Tuning for fast DB2 restart**
  - Take frequent system checkpoints
    - Every 2-5 minutes generally works well
  - Commit frequently!
  - Use DB2 Consistent restart (Postponed Abort) to limit backout of long-running URs
    - Controlled via ZPARM for normal DB2 warm restart
      - LBACKOUT=AUTO
      - BACKODUR=5 (interval is 500K log records if time-based checkpoint frequency)
    - Postponed Abort is not a 'get-out-of-jail-free' card
      - Some retained locks will persist through restart
        - Retained locks held on page sets for which backout work has not been completed
        - Retained locks held on tables, pages, rows or LOBs of those table spaces or partitions
      - If on shared critical resources, these retained locks can prevent applications from running properly

DB2 Restart Recovery …

- **Tuning for fast DB2 restart …**
  - Track and eliminate long-running URs
    - Long-running URs can have a big impact on overall system performance and availability
      - Elongated DB2 restart and recovery times
      - Reduced availability due to retained locks held for a long time (data sharing)
      - Potential long rollback times in case of application abends
      - Lock contention due to extended lock duration >> timeouts for other applications
      - Ineffective lock avoidance (data sharing)
      - Problems getting DB2 utilities executed
    - Aggressively monitor long-running URs
      - Start conservatively and adjust the ZPARM values downwards progressively
        - Long-running URs: URCHKTH=5 – based on a 3-minute system checkpoint
        - Heavy updaters that do not commit: URLGWTH = 10 (K log records)
      - Automatically capture warning messages DSNJ031I (URLGWTH) and DSNR035I (URCHKTH) and/or post process IFCID 0313 records (if Statistics Class 3 is on)
    - Get badly-behaved applications upgraded so that they commit more frequently
      - Need management ownership and process

DB2 Restart Recovery – Data Sharing Considerations

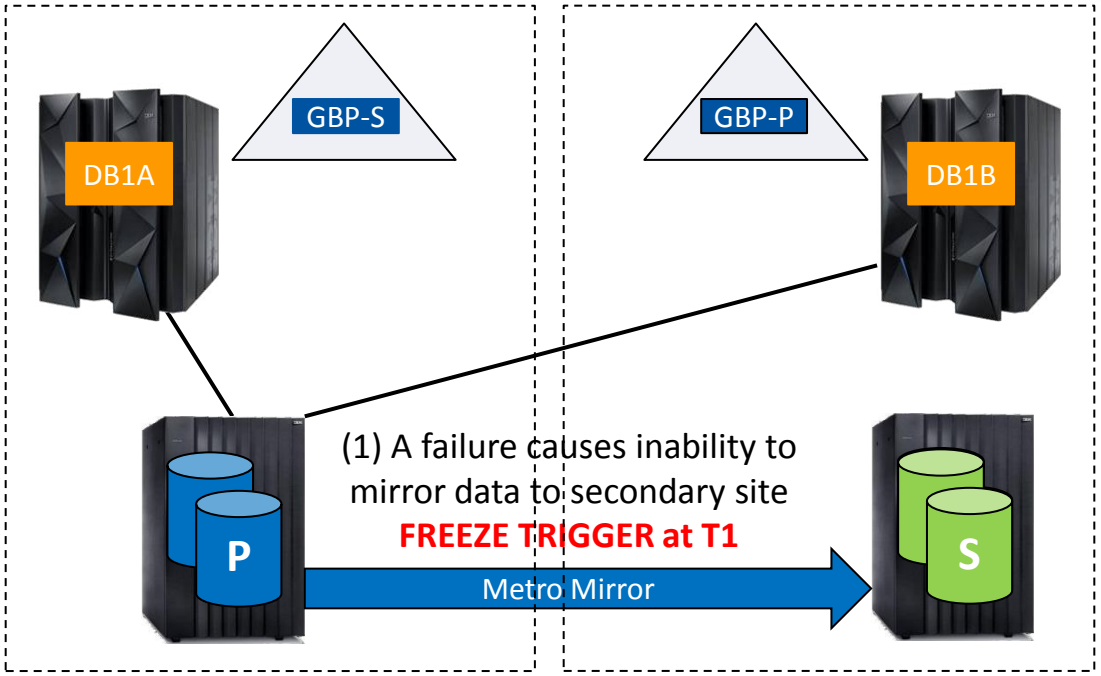
- **All DB2 CF structures existing at the DR site MUST be purged before DB2 restart**
  - Otherwise, guaranteed logical data corruption
  - Necessary actions and checks should be automated in the DR scripts

```
D XCF,STRUCTURE,STRNAME=grpname*

For any failed-persistent group buffer pools:
SETXCF FORCE,CONNECTION,STRNAME=strname,CONNAME=ALL
After the failed-persistent connection has been forced, the GBP is deallocated automatically

For the LOCK1 structure and the SCA, it is necessary to force the structures out:
SETXCF FORCE,STRUCTURE,STRNAME=strname

D XCF,STRUCTURE,STRNAME=grpname*  (to confirm)
```

  - Better approach than setting DB2 ZPARM DEL_CFSTRUCTS_ON_RESTART=YES
    - Can result in unnecessary DB2 group restart in certain local failure scenarios
  - GRECP/LPL recovery will be required as a result of deleting the GBPs
    - Likely to be the largest contributor to DB2 restart time

DB2 Restart Recovery – Data Sharing Considerations

- **Myth #6: With Metro Mirror and multi-site sysplex, no need to purge CF structures**



(2) FREEZE/GO policy: secondary devices are suspended at T1, and writes are allowed to continue on primary devices

(3) The communication failure was the first sign of a rolling DR → Site 1 is lost

No consistency between 'frozen' secondary copy and content of the CF structures

Any attempt to restart off the 'frozen' secondary copy without first purging the content of the CF structures at Site 2 will result in guaranteed data corruption

DB2 Restart Recovery – Data Sharing Considerations …

- **Optimise GRECP/LPL Recovery**
  - Tune the Group Buffer Pool thresholds to trigger frequent castout
    - Low CLASST (1-5)
      - DB2 11: CLASST can now be specified as an absolute number of pages
    - Low GBPOOLT (5-25)
    - Low GBPCHKPT (2-4)
  - Switch all objects to CLOSE YES to limit the number of objects to recover
    - Also has the potential to reduce the data sharing overhead
    - Additional recommendations
      - Tune DSMAX to avoid hitting it too often
      - Adjust PCLOSEN/T to avoid too frequent pseudo closes
        - Default PCLOSET=10 minutes is too low in data sharing
        - PCLOSET=30-45 (minutes) is usually a more suitable setting
        - Adjust PCLOSEN accordingly (V10 default is 10 system checkpoints) or disable it
        - ROT: #DSETS CONVERTED R/W -> R/O < 10-15 per minute

DB2 Restart Recovery – Data Sharing Considerations …

- **Optimise GRECP/LPL Recovery …**
  - By default, DB2 automatically initiates GRECP/LPL recovery at the end of both normal restart and disaster restart when a GBP failure condition is detected
    - Group buffer pool option AUTOREC(YES)
  - Still recommend building a home-grown GRECP/LPL recovery procedure that is optimised
    - Why do I still need a procedure for GRECP/LPL recovery?
      - If for any reason the automatic GRECP recovery fails
      - If the GBP failure occurs after DB2 restart is complete
    - How can I optimise my GRECP/LPL recovery procedure?
      - Identify the list of objects in GRECP/LPL status
      - Start with DB2 Catalog and Directory objects first
      - For the rest, generate optimal set of jobs to drive GRECP/LPL recovery
        - Spread the -START DATABASE commands across all available members
        - 51 commands maximum per member (based on 510MB of FLA storage since V10)
        - 20-30 objects maximum per -START DATABASE command

DB2 Restart Recovery – Data Sharing Considerations …

- **Optimise GRECP/LPL Recovery …**
  - Necessary actions and checks should be automated in the DR scripts
    - Check that all conditions have been resolved at the end of automatic GRECP/LPL recovery
      - Message DSNI049I - GRECP OR LPL RECOVERY FOR AUTOMATIC GRECP RECOVERY HAS COMPLETED
    - Automatically drive home-grown GRECP/LPL recovery procedure and check for completion
    - Data recovery and/or index rebuild might be required if GRECP/LPL conditions are still outstanding

      Make sure all recovery artifacts (image copies and archive logs) are available at the DR site

    - Do not start processing application workloads until all GRECP/LPL and other database exception conditions have been resolved
  - V11: Proactively track message pair DSNB355I and DSBB356I (see APAR PI22857)
    - DSNB355I - gbpname RECOVERY LRSN VALUES MIGHT CAUSE A DELAY IN GRECP RECOVERY
    - DSBB356I - CONDITION THAT CAUSED PREVIOUS DSNB355I MESSAGE DOES NOT CURRENTLY EXIST
    - Needs to be investigated as condition will elongate GRECP recovery

Testing DR

- **Need to periodically rehearse DR**
- **Objectives**
  - Validate recovery procedures and maintain readiness
  - Ensure that RTO/RPO can be achieved
  - Flush out issues and get them fixed
- **Recommendations**
  - Executing a planned site swap is an important validation but it is not enough
  - Need to stress test DB2 group restart to exercise a wide variety of conditions
    - Better understand the system behaviour and prepare for potential residual clean up actions
    - Provide a realistic service level
    - Need representative application workload running concurrently
  - Best approach: use FlashCopy technology to capture a snapshot of production at the DR site and restart off it in an isolated environment
  - Any sign of data inconsistency should be investigated and driven to root cause
    - Broken pages, data vs. index mismatches, etc.
    - Should develop procedures to proactively check for those (see next slide)

Testing DR …

- **Options to check data consistency during a DR test**
  - Consistency between table spaces and indexes
    - CHECK INDEX ALL TABLESPACE or CHECK INDEX LIST (with appropriate LISTDEF)
  - Invalid LOB values or structural problems with the pages in a LOB table space
    - CHECK LOB TABLESPACE dbname.tsname for each LOB table space
  - DB2 RI and table check constraint violations + consistency between base table and corresponding LOB/XML table spaces
    - CHECK DATA TABLESPACE dbname.tsname part SCOPE ALL
  - Options to check each page of a table space for consistency
    - DSN1COPY with CHECK option
    - SELECT * from tbname
    - Basic RUNSTATS to touch each row
  - SQL queries against DB2 catalog from migration job DSNTESQ
  - REPAIR DBD TEST or DIAGNOSE
  - For catalog objects with LOB columns:
    - CHECK LOB + CHECK INDEX on the AUX index
    - CHECK DATA on the base tablespace using SCOPE AUXONLY AUXERROR REPORT

Conclusion

- **Need consistent objectives for Continuous Availability (CA) and Disaster Recovery**
  - In line with the business requirements and expectations
  - Clearly differentiate CA and DR to ensure clarity of the objectives for functionalities

- *Running with a multi-site workload is generally done to provide faster restart in case of site failures (DR) but can compromise the exploitation of CA capabilities*
- *RPO=0 (no data loss) can only be achieved with a GDPS FREEZE/STOP or FREEZE/STOP conditionally policy that might impact the availability of production running on the primary site ('false positive')*

- **The more aggressive the SLAs, the more investments are required (escalating)**
  - Hardware (e.g. extra DASD)
  - Automated, optimised procedures
  - Testing and practice

Conclusion …

- **Data consistency is of paramount importance**
  - Any sign of inconsistency found in your testing should be driven to root cause
  - A DB2 cold start or any form of conditional restart will lead to data corruption and data loss
- **Practice, practice, practice**
  - Continually validate recovery procedures to maintain readiness
  - Verify that RPO/RTO objectives are being met
- **Do not throw away your 'standard' DB2 log-based recovery procedures**
  - Even though it should be a very rare occurrence, it is not wise to start planning for mass recovery when the failure actually occurs, e.g.
    - Plan A for Disaster Recovery has failed
    - Local recovery on the primary site following wide-spread logical corruption

Looking for More Information?

- **Redbooks**
  - *Disaster Recovery with DB2 UDB for z/OS*, SG24-6370
  - *GDPS Family - An Introduction to Concepts and Capabilities*, SG24-6374
  - *IBM DS8870 Copy Services for IBM z Systems*, SG24-6787
  - *IBM DS8870 Multiple Target Peer-to-Peer Remote Copy*, REDP-5151
  - *IBM DS8000 and z/OS Basic HyperSwap*, REDP-4441
  - *IBM Tivoli Storage Productivity Center for Replication for System z,* SG24-7563
  - *IBM z/OS Global Mirror Planning, Operations, and Best Practices,* REDP-4878

- **Manuals**
  - *z/OS DFSMS Advanced Copy Services*, SC35-0428
  - *z/OS DFSMSdfp Storage Administration,* SC26-7402
  - *z/OS DFSMShsm Storage Administration*, SC35-0421

Florence Dubois

IBM DB2 for z/OS Development

flodubois@uk.ibm.com

@floDB2z

Session M7

DB2 for z/OS and DASD-Based Disaster Recovery: Blowing Away the Myths